(19) Europäisches Patentamt
European Patent Office
Office européen des brevets

(11) EP 0 847 020 A2

(12) EUROPEAN PATENT APPLICATION

(43) Date of publication:
10.06.1998 Bulletin 1998/24

(51) Int Cl.[6]: **G06F 17/30**, G06F 12/08

(21) Application number: **97309880.9**

(22) Date of filing: **08.12.1997**

(84) Designated Contracting States:
AT BE CH DE DK ES FI FR GB GR IE IT LI LU MC NL PT SE
Designated Extension States:
AL LT LV MK RO SI

(30) Priority: **09.12.1996 US 762705**

(71) Applicant: **SUN MICROSYSTEMS, INC.**
Palo Alto, California 94303 (US)

(72) Inventors:
- **Tsirigotis, Panagiotis**
  Mountain View, CA 94040 (US)
- **Radia, Sanjay**
  Fremont, CA 94539 (US)

(74) Representative: **Harris, Ian Richard**
D. Young & Co.,
21 New Fetter Lane
London EC4A 1DA (GB)

(54) Dynamic cache preloading across loosely-coupled administrative domains

(57) A caching server that provides faster access times for independently operating network elements. The caching server initiates information transfer and holds the requested information in its memory, instead of caching information transfer in response to user requests. The caching server preloads information from another server into its memory based on a set of predetermined criteria. Such preloading preferably occurs during low usage time, such as nighttime. The information source determines which information is described in the catalog in accordance with one of a second set of predetermined criteria. For example, the catalog may be organized according to size and the information source will organize the catalog by file size. The caching server preloads as many often-used files as will fit in its cache memory

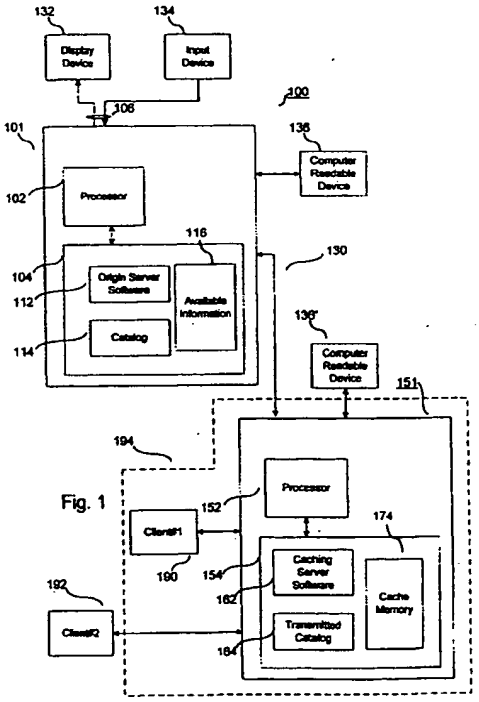

Fig. 1

EP 0 847 020 A2

Description

This application relates to networking and, specifically, to a method and apparatus for caching in a network where the elements of the network are operating independently.

Recent years have seen a huge rise in the popularity of network systems, such as the internet. A network is formed of independently operating elements that are interconnected in a way that allows them to transmit information. For example, a first network element can transmit World Wide Web pages to a second network element. The World Wide Web allows a user to use "web browser" software on his computer system to view "web pages" stored on other systems in the network. Similarly, network elements can transfer files to other network elements via File Transfer Protocol (FTP).

The amount of information transmitted over networks such as the internet has grown and will continue to grow in the future. While the overall amount of information being transferred has grown, the amount of information being transferred during a day still varies at different times of the day. A time of day when the most information is being transferred is called a "peak usage time." A time of day when a relatively small amount of information is being transferred is called a "low usage time." For example, a network that allows a consumer to download information to a personal computer using a cable modem might have a peak usage time when people get home from work and a low usage time in the middle of the night when people are asleep. It is desirable that usage of the network be evened out as much as possible. It is also desirable that response time be increased as much as possible.

Moreover, in conventional networks, various elements of the network have different goals. For example, in the above, mentioned consumer network, a centrally located network element may have a goal of sending as much information as possible out over the network. In contrast, a "local" computer may have a goal of giving "its" users the best response time possible.

Local computers often "cache" incoming information. For example, a web browser may save in its memory a copy of the ten most recent pages viewed by the user. That way, if the user wants to view any of the cached web pages, the browser software does not have to send a request for the web page out over the network. Instead, the browser can merely get the cached web page from its memory. The browser can access its memory much faster than it can request a web page and receive it over the network. Thus, caching saves the time otherwise required to fetch the web page from over the network. Local computers often cache the most recently accessed information so that the information will be available should the user request it again. A problem associated with conventional caching is that the user must wait for information to be downloaded into the local computer when the user makes a first request for the information. (Caching is performed in response to a user request for information). A second problem is that the information to be cached is often requested and downloaded during peak usage times.

Particular and preferred aspects of the invention are set out in the accompanying independent and dependent claims. Features of the dependent claims may be combined with those of the independent claims as appropriate and in combinations other than those explicitly set out in the claims.

The present invention overcomes the problems and disadvantages of the prior art by implementing a caching server that provides faster access times for independently operating network elements. Such independently operating network elements are said to have "loosely coupled administrative domains." In a preferred embodiment of the present invention, the caching server initiates information transfer and holds the requested information in its memory, instead of caching information transfer in response to user requests.

In a preferred embodiment of the present invention, the caching server preloads information from another server into its memory based on a set of predetermined criteria. For example, the caching server may download as many of the largest files as will fit in its cache memory. Such preloading preferably occurs during low usage time. Specifically, the caching server requests a "catalog" of available information from an information source, such as another server. The information source sends the catalog over the network. The caching server then decides which information it wants to preload and requests that information from the information source. Once the information is received, the caching server holds the information in a cache memory so that the information will be available when requested by a user.

Cache preloading has two main benefits: 1) it eliminates the latency time associated with accessing information requested by a user for the first time and 2) the information fetch between the information source and the caching server can take place during off-peak hours, thereby reducing the network load at peak times.

In a preferred embodiment of the present invention, the information source determines which information is described in the catalog in accordance with one of a set of predetermined criteria. For example, the catalog may be organized according to size and the information source will organize the catalog by file size. As a second example, the catalog may be organized according to the files most-often downloaded by users. In this case, the caching server will preload as many often-used files as will fit in its cache memory. As a third example, the catalog may be organized according to which files advertisers have paid to place in the catalog. In this case, the caching server will preload as many of these files as will fit in its cache memory.

In accordance with an aspect of the invention, there is provided a method for quickly accessing information in a network that includes an origin system connected

to a caching server, the method comprising the steps, performed by the caching server of: requesting a catalog from the origin system that lists documents available from the origin system; receiving the catalog from the origin system; requesting a document described in the catalog, the document requested according to a predetermined criterion of the caching server; receiving the document from the origin system; and storing the document in a cache memory of the caching server.

In accordance with another aspect of the invention, there is provided a caching server having a cache memory, the caching server being part of a network that includes an origin system connected to the caching server, the caching server comprising: a first portion configured to request a catalog from the origin system that lists documents available from the origin system; a second portion configured to receive the catalog from the origin system; a third portion configured to request a document described in the catalog, the document requested according to a predetermined criterion of the caching server; a fourth portion configured to receive the document from the origin system; and a fifth portion configured to store the document in the cache memory.

Advantages of the invention will be set forth in part in the description which follows and in part will be obvious from the description or may be learned by practice of the invention. The objects and advantages of the invention will be realized and attained by means of the elements and combinations particularly pointed out in the appended claims and equivalents.

Exemplary embodiments of the invention are described hereinafter, by way of example only, with reference to the accompanying drawings, in which:

Figure 1 is a block diagram of a data processing system in accordance with a preferred embodiment of the present invention.

Figure 2 is a diagram showing steps of a method for preloading a caching server.

Figure 3 is a diagram showing steps of a method for fulfilling a user's request for information.

Figures 4a through 4c show examples of possible catalog formats.

Reference will now be made in detail to the preferred embodiments of the invention, examples of which are illustrated in the accompanying drawings. Wherever possible, the same reference numbers will be used throughout the drawings to refer to the same or like parts.

Figure 1 is a block diagram of a computer network 100 in accordance with a preferred embodiment of the present invention. Computer network 100 includes an origin computer system 101 and a caching computer system 151. Origin system 101 and caching system 151 are connected via connection 130. The computer system 100 can be any type of network, such as the internet, a LAN, a WAN, an intranet, etc., that allows systems 101 and 151 to communicate with each other. Origin system 101 and caching system 151 operate independently and communicate using any known protocol, such as Hyper Text Transfer Protocol (http). Although only two systems 101 and 151 are shown, it should be understood that additional systems can be included in computer network 100 without departing from the the present invention.

Origin computer system 101 includes a processor 102 and a memory 104. Memory 104 includes origin server software 112 and a catalog 114. Catalog 114 contains, for example, a description of information available from origin system 101.. System 101 preferably connects to a display device 132 and an input device 134, which can be any of a preferably connects to a display device 132 and to an input device 134, which can be any of a wide range of varying I/O devices, such as disk drives, keyboards, modems, network adapters, printers, and displays. Respective computer readable devices 136, such as a disk drive or CD ROM drive, are shown connected to origin system 101 and caching system 151. Origin server software 112 and Caching server software 162 are preferably loaded into memory via devices 136.

Caching computer system 151 includes a processor 152 and a memory 154. Memory 154 includes caching server software 162 and a catalog 164 transmitted from origin system 101. Figure 1 also shows a cache memory 174. Cache memory 174 may be a part of memory 104 (as shown) or may be a separate memory. Cache memory 174 has a predetermined maximum size. In the described embodiment, caching system 151 is connected to client#1 190 and client#2 192. In the described embodiment, client#1 190 and caching system 151 jointly form a caching server 194. In other embodiments, client#1 may be located in either the same or a different computer as caching system 151. In still other preferred embodiments, caching system 151 performs the functions of client#1 190. Client#2 may be located in either the same or a different computer as caching system 151.

Preloading of caching system 151 is allowed from origin system 101, where the priorities of the entities administrating the systems 101 and are 151 different. For example, the origin system 101 will want to "push out" as much information as possible, while the caching system 151 is willing only to cache a certain amount of information. Origin system 101 may be, for example, a large, centrally located server that stores a large number of http documents, such as web pages and files. Caching system 151 may be, for example, a regional server accessible by clients in a geographical region. A typical network may include multiple origin systems 101 and/or multiple caching systems 151.

Figure 2 is a diagram showing steps of a method for preloading caching system 151 in accordance with a preferred embodiment of the present invention. The figure is divided into three areas, representing, respectively, Client#1 190, caching system 151 and origin system 101. In the described embodiment, the client and

the servers communicate with one another via the http protocol, although other suitable protocols could be used. Processor 152 executes caching server software 162 to perform the steps of the middle column. Processor 102 executes origin server software 112 to perform the steps of the right-most column. Client#1 190 and client#2 192 preferably contain a processor executing software performing the illustrated steps. As discussed above, Figure 1 shows client#1 190 as a separate process, which may or may not run on a separate machine. In an alternate embodiment, the functions of client#1 190 are performed by caching server software 162.

In step 202, client#1 190 initiates a request for a catalog to caching system 151. In the described embodiment, client#1 190 sends such a request at a predetermined time, such as in the middle of the night. It is desirable that the request of step 202 be performed at a time when the network is not busy. Such timing helps to reduce network load at peak times. In step 204, caching system 151 sends a request for a catalog to origin system 101. In step 206, origin system 101 sends the catalog 114 to caching system 151. In step 208, caching system 151 sends the catalog to client#1 190.

Steps 210 through 220 are performed multiple times until client#1 190 determines that cache memory 174 is full (or that some maximum amount of cache space is filled). Alternately, client#1 190 may send a list of information to preload to caching system 151 and caching system 151 will load as many of the items on the list as will fit in cache memory 174. In step 210, client#1 190 requests an item in the catalog from caching system 151. Caching system 151 sends the request to origin system 101, which returns the requested item. Caching system 151 caches the returned item in cache memory 174. Figure 2 shows that the item is then forwarded to client#1 190, which discards it, but may keep information about the item, such as the size. Alternately, caching system 151 may simply cache the item without sending it to client#1 190. Thus, at the end of step 216, information has been preloaded into caching system 151 and is awaiting user requests for the information.

Figure 2 also shows a step 230 in which client#1 190 sends special instructions to caching system 151. These special instructions may, for example, specify that certain information in cache memory 174 should not be discarded from cache memory for a predetermined amount of time. As another example, the special instruction may instruct caching system 151 to immediately discard certain information in cache memory 174. The information may be specified, for example, by its URL (Uniform Resource Locator). Alternately, the special instruction may instruct the caching system 151 to delete (or save) information in a last-in, first-out order or in some other appropriate order.

Figure 3 is a diagram showing steps of a method for fulfilling a user's request for information. The steps of Figure 3 are performed after the caching system 151 has preloaded some information into cache memory 174, as shown in Figure 2. In the example, the user is logged onto client#2 192. In step 302, the user requests information such as a file or a web page by URL. If, in step 304, the requested information has been preloaded and exists in cache memory 174, the information is copied from cache memory 174 and sent to client#2 192. If, in step 308, the information was not preloaded, then, in step 310, caching system 151 requests the information from origin system 101, which returns the requested information. In step 314, caching system 151 stores the information in cache memory 174 and, in step 316, sends the information to the requesting client. Caching system 151 determines whether certain information is present in cache memory 174 via any of a variety of known cache memory techniques.

Note that step 314 may involve purging (discarding) items from cache 174 to make room for new items. If caching system 151 has received special instructions not to discard certain cached information, then in step 314, that information will not be discarded from cache, even if it would normally be next in line to be discarded.

Figures 4a through 4c show respective examples of formats of catalog 114. Each example format includes a section or row in the catalog for each file or URL that is available to be downloaded from the origin system. These formats are provided by way of example only and are not meant to limit the scope of the present invention. Catalog 114 is preferably an http document of type "text/plain." Figure 4a shows a format in which catalog 114 includes two fields: a Uniform Resource Locator (URL) field 402 and a size field 404. The catalog may optionally be organized by increasing or decreasing size.

Figure 4b shows a format in which catalog 114 includes three fields: a URL field 410, a size field 412, and a "# of times downloaded" field 414 (also called a "popularity" field). Alternately, the popularity field 414 could be omitted and the catalog could simply be sorted by ascending or descending popularity value.

Figure 4c shows a format in which the catalog includes three fields: a URL field 420, a size field 412, and a "desirability field." Alternately, the desirability field 412 could be omitted and the catalog could simply be sorted by ascending or descending desirability. In this example, desirability indicates a criteria defined in origin system 101. For example, if an advertiser had paid a premium to have his information placed into the catalog, the information would have a high desirability value in the catalog. Other examples of catalogs might have the URLs sorted according to their PICS rating, or any similar rating which rates the content of files.

Caching system 151 reviews the contents of catalog 114 and determines which available information described in catalog 114 it should preload. This determination is made in accordance with a number of criteria stored in the memory of caching system 151. A simple criteria, for example, is to simply preload files in the catalog by size, largest first, until cache memory 174 is full (or a predetermined cache size limit has been reached).

Another possible criteria might be that caching system 151 preloads according to both the size of the file, the available cache space, and the "popularity" or "desirability" of the file. Another possible criteria might be that the caching system 151 preloads in accordance with one of the criteria described above, but does not, under any circumstances, download information on a "do not download" list stored in its memory. For example, certain caching systems 151 might never want to download adult material or material from a predetermined information source. As can be seen from the above examples, the criteria used to determine what information to download is affected somewhat by what information is available in catalog 114. A large number of catalog formats and caching system criteria are compatible with embodiments of the present invention.

In some embodiments, origin system 101 gives "hints" to the caching system/client as to when it should next request a catalog. Theses hints may be sent as part of the catalog or may be sent in a separate administrative message. For example, if the catalog is normally updated every twenty-four hours, origin system 101 might suggest that the catalog be downloaded by the caching system every twenty-four hours. If, however, some important event is occurring, origin system 101 might suggest that the catalog be downloaded more frequently because it is being updated more frequently by origin system 101.

In a preferred embodiment, if origin system 101 is accessed by multiple caching servers, a different catalog can be sent to each caching server 194. The catalog is dynamically generated based on, for example, the access pattern generated by the particular caching server as observed by the origin system. Alternately, catalogs are determined by policy decisions at the origin system 101. For example, the catalogs sent to a certain caching server may not contain adult material.

In summary, an embodiment of the invention allows a caching server, preferably including a client and a caching system, to initiate a request for a catalog of available http documents. This request is made at night or during a slow time on the network. After the catalog is received, the caching server decides which of the documents in the catalog to preload into a cache memory. Thus, the preloaded documents will be available in the cache memory of the caching server when a user requests them. The caching server can decide that some or all preloaded documents should not be purged from its cache for a predetermined time.

Other embodiments will be apparent to those skilled in the art from consideration of the specification and practice of the invention disclosed herein. It is intended that the specification and examples be considered as exemplary only.

Claims

1. A method for accessing information in a network that includes an origin system connected to a caching server, the method comprising the steps, performed by the caching server of:

   requesting a catalog from the origin system that lists documents available from the origin system;
   receiving the catalog from the origin system;
   requesting a document described in the catalog, the document requested according to a predetermined criterion of the caching server;
   receiving the document from the origin system; and
   storing the document in a cache memory of the caching server.

2. The method of claim 1, further including the steps of:

   receiving, by the caching server, a request for the document from a user; and
   sending, by the caching server, the document stored in the cache memory to the user.

3. The method of claim 1, wherein the step of requesting a catalog from the origin system performed during a low usage time of the network.

4. The method of claim 1, wherein the caching server includes a client and a caching system and wherein the step of requesting a catalog from the origin system includes the steps of:

   sending the request for the catalog from the client to the caching system; and
   sending the request for the catalog from the caching system to the origin system.

5. The method of claim 1, wherein the caching server includes a client and a caching system and wherein the step of requesting a document described in the catalog includes the steps of:

   sending the request for the catalog from the client to the caching system; and
   sending the request for the catalog from the caching system to the origin system.

6. The method of claim 1, wherein the criterion used by the caching server to determine whether to request a document from the catalog is based on the size of the document.

7. The method of claim 1, wherein the criterion used by the caching server to determine whether to re-

quest a document from the catalog is based on a rating associated with the document and stored in the catalog.

8. The method of claim 1, wherein the criterion used by the caching server to determine whether to request a document from the catalog is based on a list of non-acceptable document names.

9. The method of claim 1, wherein the network includes a second caching server and the first and second caching servers receive different catalogs from the origin system.

10. The method of claim 1, wherein the network includes a second caching server and the first and the second caching servers request different documents from the origin system.

11. The method of claim 1, wherein the origin system and the caching server use http protocol to communicate.

12. The method of claim 1, wherein the caching server includes a client and a caching system and further including the steps of:

sending, by the client to the caching system, an instruction to retain the document in the cache memory for a specified amount of time; and
retaining the document in the cache memory, by the caching system, for the specified amount of time.

13. The method of claim 1, wherein the caching server includes a client and a caching system and further including the steps of:

sending, by the client to the caching system, an instruction to purge the document from the cache memory; and
purging the document from the cache memory, by the caching system.

14. A caching server having a cache memory, the caching server being part of a network that includes an origin system connected to the caching server, the caching server comprising:

a first portion configured to request a catalog from the origin system that lists documents available from the origin system;
a second portion configured to receive the catalog from the origin system;
a third portion configured to request a document described in the catalog, the document requested according to a predetermined criterion of the caching server;
a fourth portion configured to receive the document from the origin system; and
a fifth portion configured to store the document in the cache memory.

15. The caching server of claim 14, wherein the caching server includes a client and a caching system and wherein the first portion includes:

a sixth portion configured to send the request for the catalog from the client to the caching system; and
a seventh portion configured to send the request for the catalog from the caching system to the origin system.

16. A computer program product comprising:

a computer usable medium having computer readable code embodied therein for causing access to information stored on a network having a origin system and a caching server, the computer program product comprising:
computer readable program code devices configured to cause a computer to effect requesting a catalog from the origin system that lists documents available from the origin system;
computer readable program code devices configured to cause a computer to effect receiving the catalog from the origin system;
computer readable program code devices configured to cause a computer to effect requesting a document described in the catalog, the document requested according to a predetermined criterion;
computer readable program code devices configured to cause a computer to effect receiving the document from the origin system; and
computer readable program code devices configured to cause a computer to effect storing the document in the cache memory.

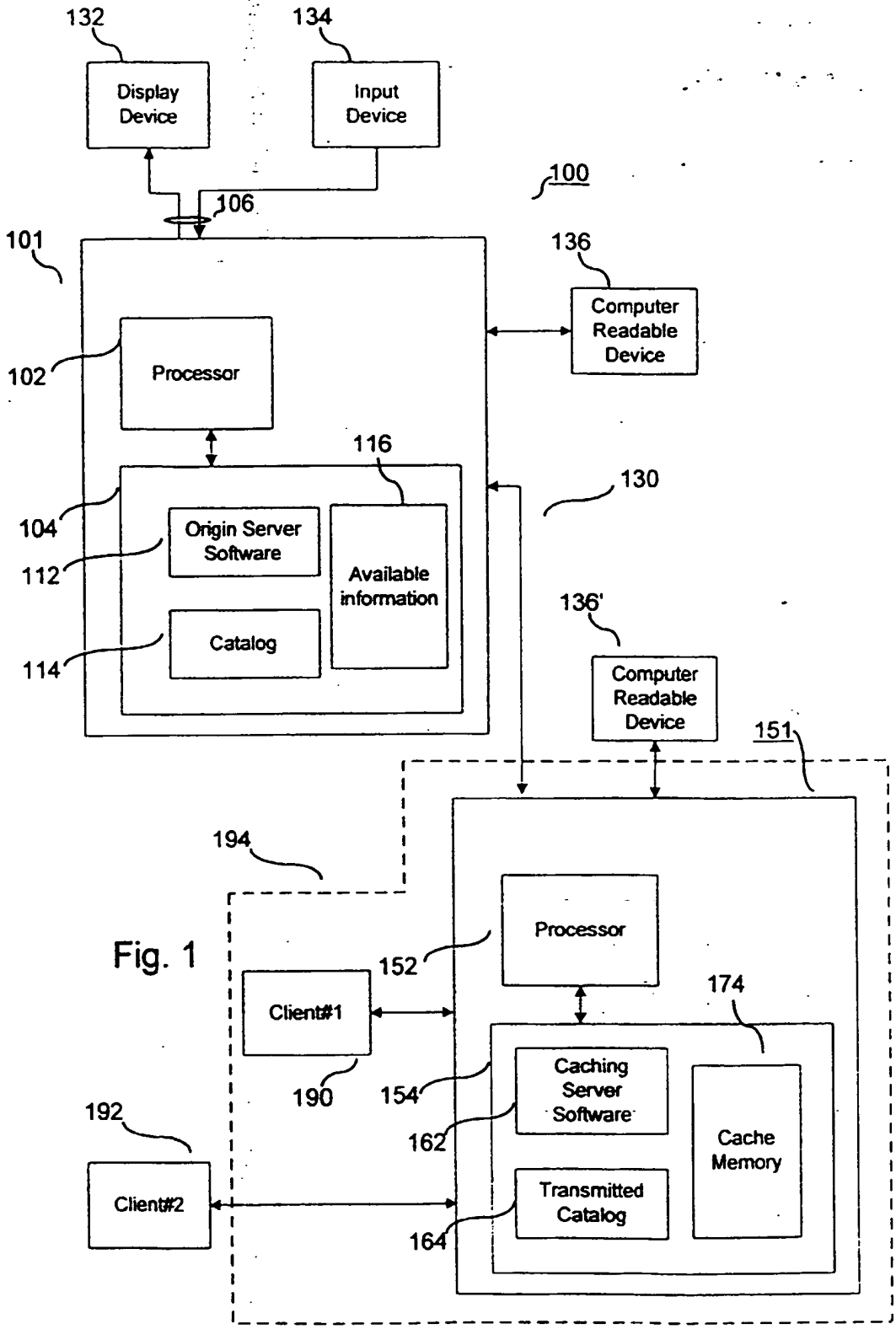
132
Display Device
134
Input Device
100
106
101
136
Computer Readable Device
102
Processor
116
130
104
Origin Server Software
112
Available information
136'
Catalog
114
Computer Readable Device
151
194
Processor
152
Client#1
174
Caching Server Software
190
154
Cache Memory
192
162
Transmitted Catalog
Client#2
164

Fig. 1

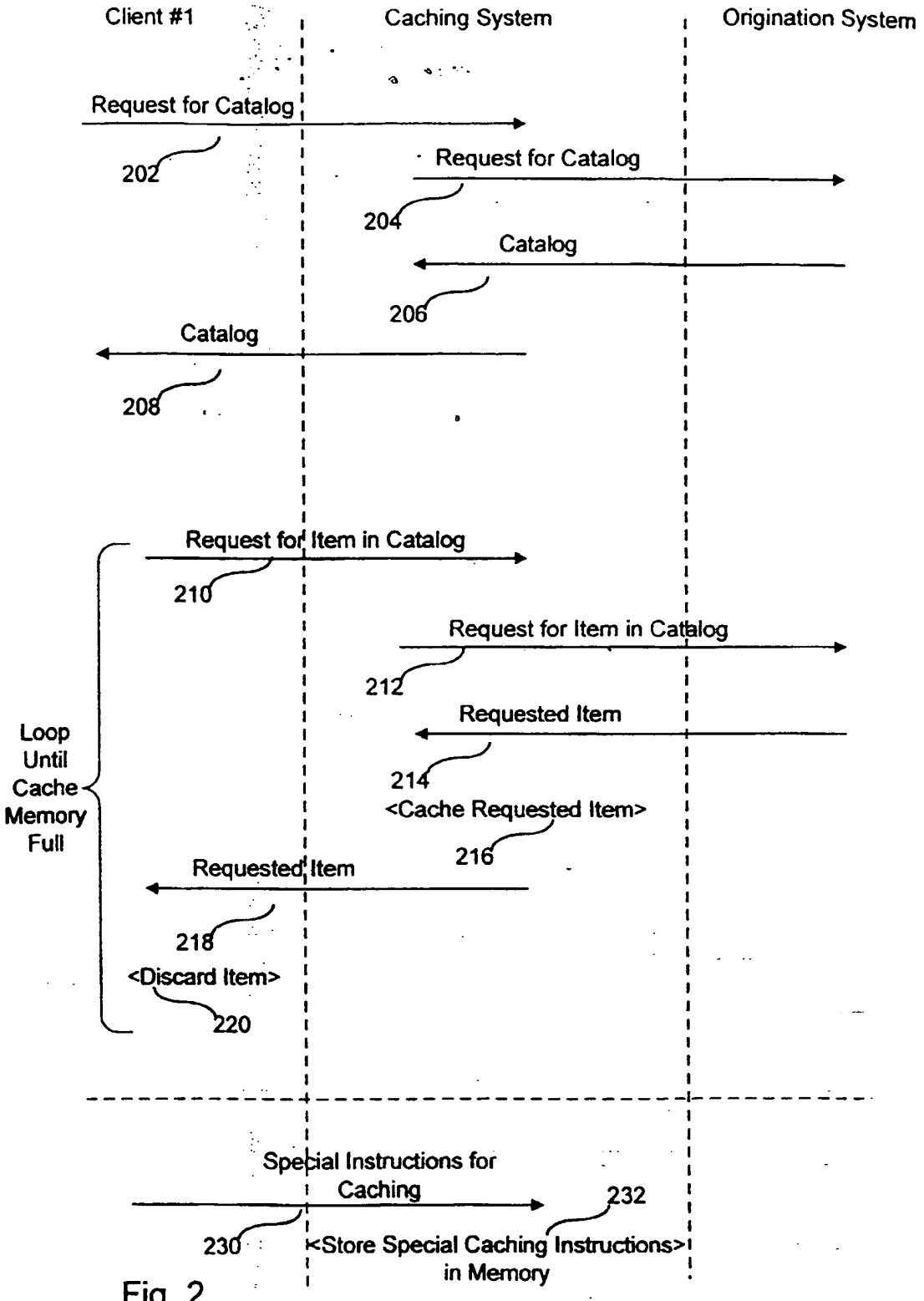
Client #1
Caching System
Origination System
Request for Catalog
202
Request for Catalog
204
Catalog
206
Catalog
208
Request for Item in Catalog
210
Request for Item in Catalog
212
Requested Item
214
<Cache Requested Item>
216
Requested Item
218
<Discard Item>
220
Loop Until Cache Memory Full
Special Instructions for Caching
230
232
<Store Special Caching Instructions> in Memory

Fig. 2

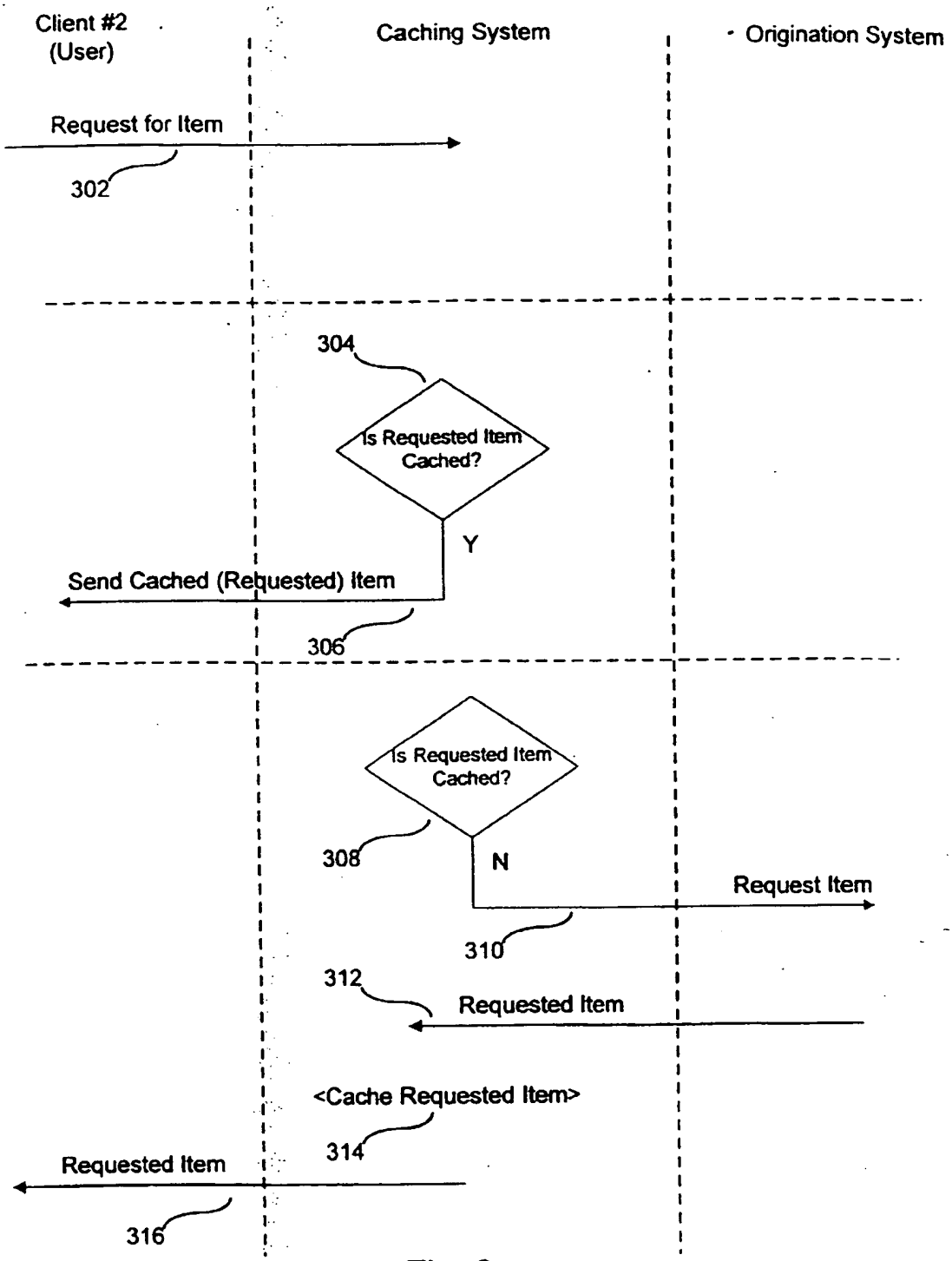
Client #2
(User)
Caching System
Origination System
Request for Item
302
304
Is Requested Item Cached?
Y
Send Cached (Requested) Item
306
Is Requested Item Cached?
308
N
Request Item
310
312
Requested Item
<Cache Requested Item>
314
Requested Item
316

Fig. 3

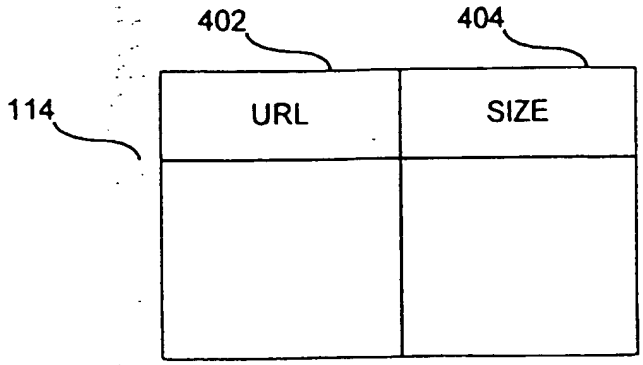

Fig. 4(a)

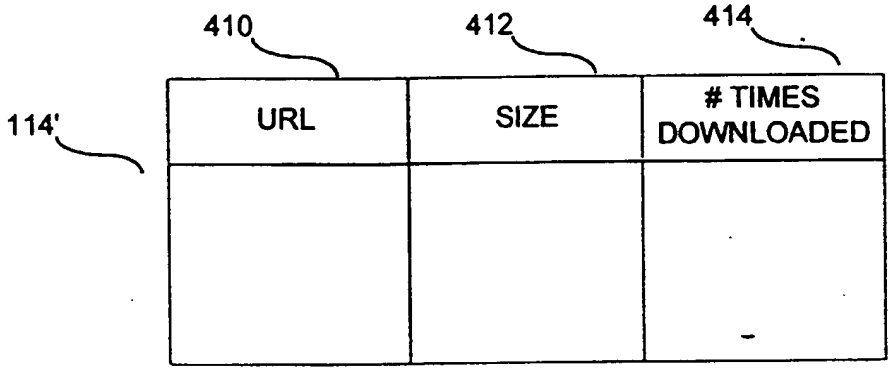

Fig. 4(b)

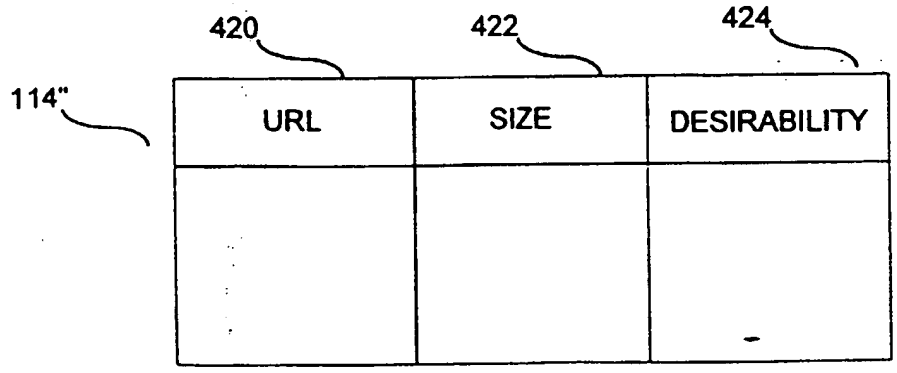

Fig. 4(c)

(19) Europäisches Patentamt
European Patent Office
Office européen des brevets

(11) EP 0 847 020 A3

(12) EUROPEAN PATENT APPLICATION

(88) Date of publication A3:
21.04.1999 Bulletin 1999/16

(43) Date of publication A2:
10.06.1998 Bulletin 1998/24

(21) Application number: **97309880.9**

(22) Date of filing: **08.12.1997**

(51) Int Cl.[6]: **G06F 17/30**, G06F 12/08, H04L 29/06

(84) Designated Contracting States:
AT BE CH DE DK ES FI FR GB GR IE IT LI LU MC NL PT SE
Designated Extension States:
AL LT LV MK RO SI

(30) Priority: **09.12.1996 US 762705**

(71) Applicant: **SUN MICROSYSTEMS, INC.**
Palo Alto, California 94303 (US)

(72) Inventors:
- **Tsirigotis, Panagiotis**
  Mountain View, CA 94040 (US)
- **Radia, Sanjay**
  Fremont, CA 94539 (US)

(74) Representative: **Harris, Ian Richard**
D. Young & Co.,
21 New Fetter Lane
London EC4A 1DA (GB)

(54) Dynamic cache preloading across loosely-coupled administrative domains

(57) A caching server that provides faster access times for independently operating network elements. The caching server initiates information transfer and holds the requested information in its memory, instead of caching information transfer in response to user requests. The caching server preloads information from another server into its memory based on a set of predetermined criteria. Such preloading preferably occurs during low usage time, such as nighttime. The information source determines which information is described in the catalog in accordance with one of a second set of predetermined criteria. For example, the catalog may be organized according to size and the information source will organize the catalog by file size. The caching server preloads as many often-used files as will fit in its cache memory

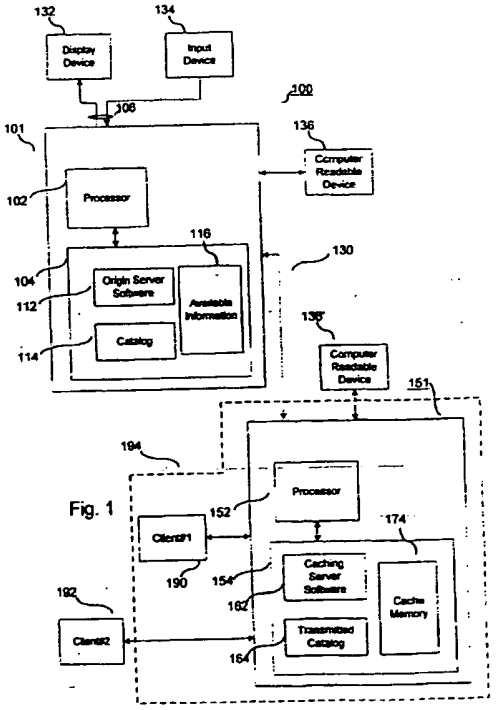

Fig. 1

EP 0 847 020 A3

European Patent Office

EUROPEAN SEARCH REPORT

Application Number
EP 97 30 9880

DOCUMENTS CONSIDERED TO BE RELEVANT

Category	Citation of document with indication, where appropriate, of relevant passages	Relevant to claim	CLASSIFICATION OF THE APPLICATION (Int.Cl.6)
A	BESTAVROS A: "SPECULATIVE DATA DISSEMINATION AND SERVICE" DATA ENGINEERING,1996, pages 180-187, XP000764867 * page 180, left-hand column, line 1 - page 181, left-hand column, line 1 * * page 186, right-hand column, line 33 - page 187, left-hand column, paragraph 4 *	1-7, 9-11, 14-16	G06F17/30 G06F12/08 H04L29/06
A	WANG Z ET AL: "PREFETCHING IN WORLD WIDE WEB" COMMUNICATIONS: THE KEY TO GLOBAL PROSPERITY. GLOBECOM 1996 GLOBAL INTERNET 96 CONFERENCE RECORD, LONDON, NOV. 18 - 22, 1996, vol. SUPP, 18 November 1996, pages 28-32, XP000741668 INSTITUTE OF ELECTRICAL AND ELECTRONICS ENGINEERS * page 29, left-hand column, paragraph 4 - right-hand column, paragraph 4.1 * * page 30, left-hand column, line 44 - page 31, left-hand column, paragraph 6 *	1-7, 9-11, 14-16	
A	SCHILIT B N ET AL: "TeleWeb: Loosely connected access to the World Wide Web" COMPUTER NETWORKS AND ISDN SYSTEMS, vol. 28, no. 11, May 1996, page 1431-1444 XP004018240 * page 1433, right-hand column, paragraph 2.1 - page 1434, left-hand column, paragraph 2.2 * * page 1437, left-hand column, paragraph 4.1 - page 1438, right-hand column, paragraph 4.2; figure 1 *	1,12-14, 16	

-/--

TECHNICAL FIELDS SEARCHED (Int.Cl.6)

G06F

The present search report has been drawn up for all claims

Place of search	Date of completion of the search	Examiner
THE HAGUE	22 February 1999	Fournier, C

CATEGORY OF CITED DOCUMENTS

X : particularly relevant if taken alone
Y : particularly relevant if combined with another document of the same category
A : technological background
O : non-written disclosure
P : intermediate document

T : theory or principle underlying the invention
E : earlier patent document, but published on, or after the filing date
D : document cited in the application
L : document cited for other reasons

& : member of the same patent family, corresponding document

EPO FORM 1503 03.82 (P04C01)

European Patent Office

EUROPEAN SEARCH REPORT

Application Number
EP 97 30 9880

DOCUMENTS CONSIDERED TO BE RELEVANT

Category	Citation of document with indication, where appropriate, of relevant passages	Relevant to claim	CLASSIFICATION OF THE APPLICATION (Int.Cl.6)
A	GWERTZMAN J S ET AL: "The case for geographical push-caching" PROCEEDINGS FIFTH WORKSHOP ON HOT TOPICS IN OPERATING SYSTEMS (HOTOS-V) (CAT. NO.95TH8059), PROCEEDINGS 5TH WORKSHOP ON HOT TOPICS IN OPERATING SYSTEMS (HOTOS-V), ORCAS ISLAND, WA, USA, 4-5 MAY 1995, pages 51-55, XP002094210 ISBN 0-8186-7081-9, 1995, Los Alamitos, CA, USA, IEEE Comput. Soc. Press, USA * the whole document *	1,14,16	

TECHNICAL FIELDS SEARCHED (Int.Cl.6)

The present search report has been drawn up for all claims

Place of search	Date of completion of the search	Examiner
THE HAGUE	22 February 1999	Fournier, C

CATEGORY OF CITED DOCUMENTS

X : particularly relevant if taken alone
Y : particularly relevant if combined with another document of the same category
A : technological background
O : non-written disclosure
P : intermediate document

T : theory or principle underlying the invention
E : earlier patent document, but published on, or after the filing date
D : document cited in the application
L : document cited for other reasons

& : member of the same patent family, corresponding document

EPO FORM 1503 03.82 (P04C01)